APPROVAL SHEET

Title of Thesis: ANALYSIS OF SECURITY TOOLS IN CLOUD COMPUTING

Name of Candidate: Aishwarya Murumkar

Master of Science in Computer Science

Fall 2020

Thesis and Abstract Approved: 

Charles K. Nicholas, Ph.D.

Professor

Department of Computer Science

and Electrical Engineering

Date Approved: November 27, 2020

ABSTRACT

Title of thesis: Analysis of Security tools in Cloud Computing
MASTER'S THESIS

Aishwarya Murumkar, Master of Science, 2020

Thesis directed by: Professor Charles Nicholas
Department of Computer Science and
Electrical Engineering

Cloud computing is a new computational standard which offers an innovative business model for organizations to adopt IT without upfront investment. Regardless of the potential gains achieved from the cloud computing, the model security is still debatable which impacts the cloud adoption. The security problem becomes more complicated under the cloud model as new dimensions have entered into the problem such as scalability, manageability and cost. Cloud service users need to be vigilant in understanding the risks of data breaches in this new environment.

In this paper, we discuss existing know problems and vulnerabilities in the cloud and investigate two highly efficacious Cloud platforms: Microsoft Azure(Azure) and Amazon Web Services(AWS) security tools and their adoption and integration with existing security tools.

We conclude with a discussion as to which Cloud Computing technology, Microsoft Azure or AWS, is better to adopt in terms of security and cost.

Analysis of Security tools in Cloud Computing

by

Aishwarya Girish Murumkar

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, Baltimore County in partial fulfillment
of the requirements for the degree of
Master of Science
2020

Advisory Committee:
Professor Charles Nicholas, Chair/Advisor
Professor Maya Larson
Dr. Edward Zieglar

Acknowledgments

I would like to express appreciation for several people who have helped and supported my degree and this thesis work. First, I would like to thank my thesis supervisor and committee chair, Dr. Charles Nicholas who offered valuable advice, support, and feedback throughout my master's thesis research. Secondly, I would like to thank Dr. Maya Larson and Dr. Edward Zieglar for their participation on the thesis committee, and for their evaluation and feedback. Thank you.

Table of Contents

List of Figures

List of Abbreviations

AWS	Amazon Web Service
CSP	Cloud Service Provider
CVE	Common Vulnerabilities and Exposure
CIS	Centre of Internet Security
EC2	Elastic Compute Cloud
VPC	Virtual Private Cloud
IAM	Identity and Access Management
NIST	National Institute of Standards and Technology (U.S.)
CC	Cloud Computing
OWASP	Open Web Application Security Project

Chapter 1: Introduction

The most widely used definition of the cloud computing model is introduced by NIST [1] as "a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g.,networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.".

1.1 Cloud Computing aspects and features

Cloud computing (CC) gained a widespread acceptance as a paradigm of computing. The main aim of CC is to reduce the need for customers' investment in new hardware or software by offering flexible cloud services, with a user reaping the benefits of the pay per use approach. CC demands addressing many security and privacy issues: both problems (vulnerabilities, threats, and attacks) and solutions (controls). The characteristics defined are as follows

1. **On-demand self-service**: The customer can choose the resources they want for their use as needed without separate agreements.

2. **Broad network access**: The capabilities are available using thin or thick

client platforms through networks. These platforms include e.g. mobile phones, tablets, laptops and workstations.

3. **Resource pooling**: The service provider's computing resources are shared among several customers. The customer can specify the region where the selected resources are used.

4. **Rapid elasticity**: The capabilities are automatically scaled according to demand and are available anywhere, anytime.

5. **Measured service**: The resources are configurable to respond as desired when they reach a certain measurable threshold, and their changes can be monitored as needed. For example, if the storage capacity of the database reaches 80% of the maximum, then the database expansion will be triggered automatically.

Mell et al. (2011, 2-3) [2] have defined service models as follows.

1. **Software as a Service (SaaS)**: Consumer can access the applications provided by the cloud provider through either a thin client interface or a program interface. The service provider manages the underlying cloud infrastructure.

2. **Platform as a Service (PaaS)**: Service provider manages the underlying cloud infrastructure where the customer can deploy the applications, they create that are made with the tools provided by the service. The customer manages their own applications and possibly the configuration of the applications.

3. **Infrastructure as a Service (IaaS)**: Service provider manages the cloud infrastructure and the customer can install applications, manage operating systems, storage, etc. This model gives the customer the most extensive access.

Mell et al. (2011, 3) [2] have defined deployment models as follows.

1. **Private cloud**: Cloud infrastructure is privately used, managed, and possibly owned by some organization. Community cloud means that the cloud infrastructure is intended for use by a specific community of concern.

2. **Public Cloud**: Cloud infrastructure is intended for public use, it is under the control and premises of the service provider.

3. **Hybrid cloud**: Cloud infrastructure is a composition of the above-mentioned cloud infrastructures. These clouds are configured with each other so that the transferability of data and applications between them is enabled.

1.2 Cloud Service Providers

Defined by the International Standards Organization, a Cloud Service Provider (CSP) is a party which makes cloud services available. A CSP focuses on activities necessary to provide a cloud service and to ensure its delivery to the customer. These activities include, not exhaustively, deploying and monitoring the service, providing audit data and maintaining the infrastructure.

1. **Amazon Web Services** AWS is a cloud service platform that offers SaaS, PaaS and IaaS with highly reliability, scalability and low-cost infrastructure.

AWS was officially launched in 2006. Within 12 geographic Regions worldwide, AWS operates in 33 Availability Zones. Data center locations [3] are in U.S., Europe, Brazil, Singapore, Japan, and Australia. About 11 more Availability Zones and 5 regions are expected to come online [4]. Elastic Compute Cloud (EC2) from Amazon, virtual private cloud (VPC), Rout 53 (a highly available and scalable cloud Domain Name System (DNS) web service), Relational Database Service (RDS), Elastic load balancer (ELB), Simple Storage Service (S3), Elastic Block Store (EBS), Glacier, Simple Queue Service (SQS)/ Auto Scale, Security Group and Cloudfront are some of the services provided by AWS [5].

2. **Microsoft Azure** Azure is a popular cloud service platform and infrastructure; it offers SaaS, PaaS and IaaS with highly reliability, scalability and low-cost infrastructure. Azure was first launched in 2008. It is available in 140 countries, including China, and supports 10 languages, 24 currencies, and the data centers available [6] in 28 regions [7]. Some of services that Azure offers to customers are Virtual Machine, Virtual Network, Windows Azure Name Resolution, Structure Query Language (SQL) Database, Traffic Manager, Storage, Scheduler, EndPoint and Content Delivery Network (CDN) [8].

Salesforce has been a pioneer in introducing cloud computing to the public by delivering enterprise applications over the Internet since 1999 [9]. Initially as a subsidiary of Amazon.com, Amazon Web Services (AWS) [10] entered the market in 2006 with the release of their Elastic Compute Cloud (EC2). Around 2010, Google

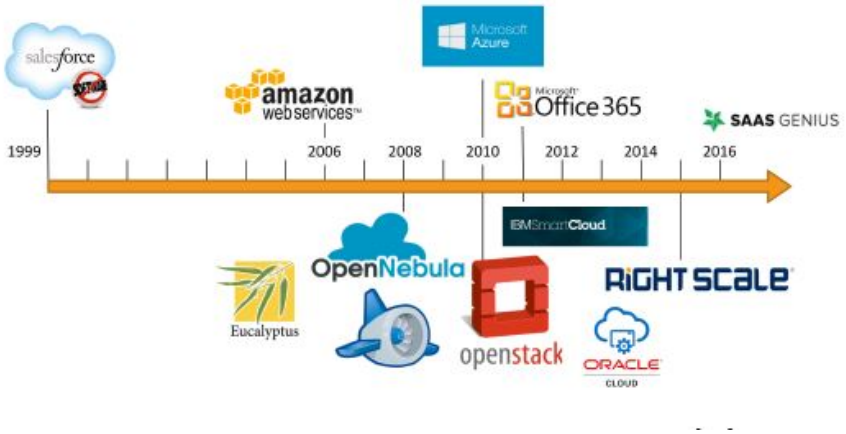

Figure 1.1: Timeline of cloud service providers [1]

and Microsoft began to invest in this area as well.

1.3 Shared responsibility model

The key to a successful security implementation in a cloud environment is understanding where your provider's responsibility ends, and where yours begins. The answer isn't always clear-cut, and definitions of the shared responsibility security model can vary between service providers and can change based on whether you are using infrastructure-as-a-service (IaaS) or platform-as-a-service (Paas): In the AWS Shared Security [11] model below, AWS claims responsibility for "protecting the hardware, software, networking, and facilities that run AWS Cloud services."

Microsoft Azure claims security ownership of "physical hosts, networks, and data centers." Both AWS and Azure state that your retained security responsibilities depend upon which services you select. While the wording is similar, shared

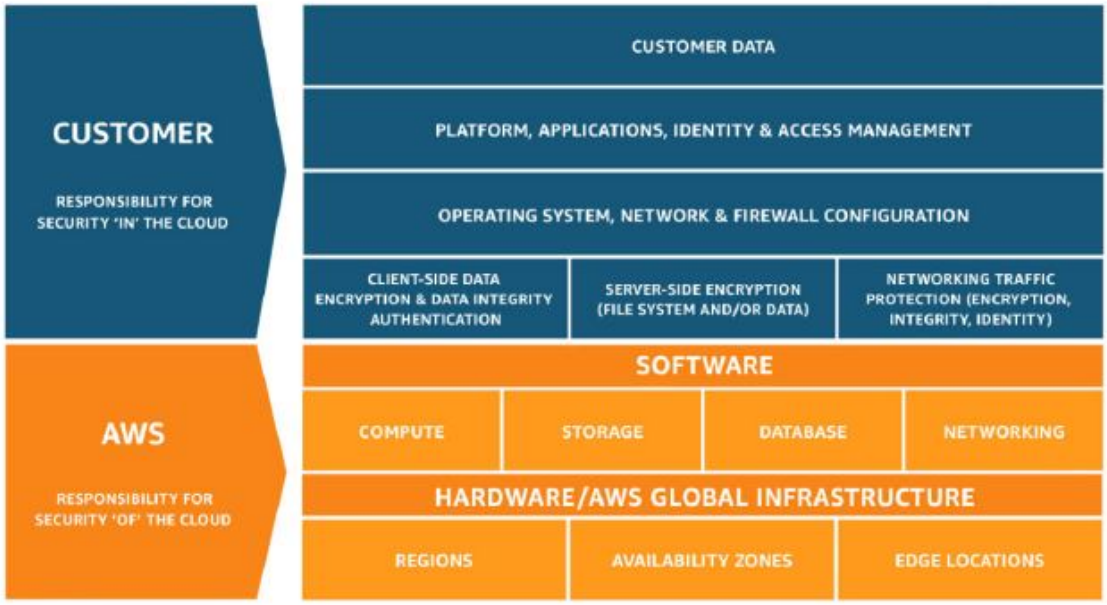

Figure 1.2: Shared Responsibility Model

responsibility agreements leave much open for discussion and interpretation. But there are always some aspects of security that are clearly owned by the provider and others that you will always retain. For the services, applications, and controls between those ownership layers, security responsibilities vary by cloud provider and service type. In a multi-cloud environment, these variations in ownership introduce complexity and risk. Each environment, application, and service requires a unique approach for security assessment and monitoring. However, your overall security posture is defined by your weakest link. If there is a gap in coverage in any one system, we increase vulnerability across the entire stack and out to any connected systems.

Although there are many benefits to adopting Cloud Computing, there are also some significant barriers to adoption. One of the most significant barriers

to adoption is security, followed by issues regarding compliance, privacy and legal matters [12]. Because Cloud Computing represents a relatively new computing model, there is a great deal of uncertainty about how security at all levels (e.g., network, host, application, and data levels) can be achieved and how applications security is moved to Cloud Computing [13]. That uncertainty has consistently led information executives to state that security is their number one concern with Cloud Computing [14].

THESIS STATEMENT This thesis describes categorization of security issues of cloud computing by two different cloud service providers which assesses and exhibits how Amazon and Microsoft implemented and structured their cloud infrastructure to tackle these issues. Also, it provides the criteria for accessing the suitability of a business or organisation for choosing a cloud provider to secure their cloud.

1.4 Outline of Thesis

This thesis is organized as follows: Chapter 2, presents the related work in cloud computing security such as vulnerability scans and penetration tests performed on cloud and it describes the history of issues and threats in two major cloud providers. Chapter 3, shows the experiments performed using in-house cloud tools used in those CSP's to scan their systems. Chapter 4, describes the findings of the experiments that were carried out and there efficiency. Chapter 5, summarizes those findings and suggests areas for future work.

Chapter 2: Related Work

2.1 Overview of existing, known problems in AWS and Azure

As described in Chapter 1, Cloud Computing leverages many existing technologies such as web services, web browsers, and virtualization, which contributes to the evolution of cloud environments. Security threats are constantly evolving and becoming more sophisticated, and cloud computing is no less at risk than an on-premise environment. A threat is a potential attack that may lead to a misuse of information or resources, and the term vulnerability refers to the flaws in a system that allows an attack to be successful. Therefore, any vulnerability associated to these technologies also affects the cloud, and it can even have a significant impact.

2.1.1 Common threats and vulnerabilities in Cloud

1. **Account or service hijacking:** An account theft can be performed in different ways such as social engineering and weak credentials. If an attacker gains access to a user's credential, he can perform malicious activities such as accessing sensitive data, manipulating data, and redirecting any transaction.

2. **Data leakage:** Data leakage happens when the data gets into the wrong

hands while it is being transferred, stored, audited or processed.

3. **Denial of Service:** It is possible that a malicious user will take all the possible resources or can block the access as well. Thus, the system cannot satisfy any request from other legitimate users due to resources being unavailable [15].

4. **Customer-data manipulation:** Users attack web applications by manipulating data sent from their application component to the server's application. For example, SQL injection, command injection, insecure direct object references, and cross-site scripting.

5. **VM escape:** It is designed to exploit the hypervisor in order to take control of the underlying infrastructure.

6. **VM hopping:** It happens when a VM is able to gain access to another VM (i.e. by exploting some hypervisor vulnerability).

7. **Malicious VM creation:** An attacker who creates a valid account can create a VM image containing malicious code such as a Trojan horse and store it in the provider repository [16].

8. **Insecure VM migration:** Live migration of virtual machines exposes the contents of the VM state files to the network. An attacker can do the following actions: a) Access data illegally during migration b) Transfer a VM to an untrusted host c) Create and migrate several VMs causing disruptions or DoS.

9. **Sniffing/Spoofing virtual networks:** A malicious VM can listen to the

virtual network or even use ARP spoofing to redirect packets from/to other VMs/.

2.1.2 Security Incidents/Issues on AWS Cloud

Amazon Web Services is considered to provide a well-secured environment in the cloud, as shown by several certificates owned by the company [17]. Nevertheless, inappropriate usage of the services can be the source of severe security breaches. In this section, those vulnerabilities are reviewed that have been identified so far and have been proven to be legitimate concerns.

In the following, previous incidents are shortly reviewed when certain errors lead to relevant security breaches.

1. **Accenture case** In 2017, four Amazon S3 buckets were discovered by Cyber Risk Research to be configured for public access [18]. As mentioned previously, all S3 buckets have a globally unique name, therefore these buckets could be bound to Accenture, a management consulting company. The buckets contained secret API data, authentication credentials, decryption keys and customer data which could have exposed the clients to serious risk. Fortunately, the publicly available storages were discovered before being accessed by anyone with malicious intent.

2. **U.S. voter records** The incident of Accenture was not the only discovery by Upguard's Cyber Risk Team. The largest data exposure of its kind made 198 million records on American voters vulnerable, including personal and

analytics data [19]. In total, the personal information of nearly all of America's 200 million registered voters was exposed, including names, dates of birth, home addresses, phone numbers, and voter registration details, as well as data described as modeled voter ethnicities and religions. The data was stored on a publicly accessible S3 storage server owned by a Republican data analytics company, Deep Root Analytics. Due to a responsible disclosure, the server was secured prior to any publication.

3. **AgentRun case** Health and medical data is always considered to be among the most confidential. AgentRun, customer management software for insurance brokers, accidentally exposed personal and medical information on thousands of customers of major insurance companies [20]. During an application upgrade, they migrated to an S3 bucket in which configurations were not handled cautiously. The bucket contained sensitive health information such as individuals prescriptions, dosages, costs, and personal data, in some cases including income range or ethnicity.

These three cases above are only a slight selection of the several incidents that took place in the past. The collection of Peter Benjamin called **YAS3BL** (Yet Another S3 Bucket Leak) lists all preceding S3 bucket leaks that have been discovered and made public [21]. At the time of this writing, 27 previous cases are listed with the number of records involved and the type of data that has been leaked.

4. **IAM policy misuse** IAM is the core service behind access management within the AWS environment. For this reason, misconfigurations of the service is the main source of vulnerabilities, once an EC2 instance is compromised. The misuse of IAM policies and permissions can lead to privilege escalation or data exfiltration. In fact, the previously mentioned S3 bucket vulnerability can be a consequence of IAM policy misuse as well.

2.1.3 Security Incidents/Issues in Azure Cloud

1. **Azure Blob Storage Is Common Target Of Hackers** Azure has been abused a bit more than AWS in actual attacker stagecraft since it is a trusted environment that can be set up for free. That's expected to continue going forward, according to Ryan Kalember, Proofpoint's EVP of cybersecurity strategy. Attackers are very familiar with the Microsoft ecosystem, Kalember said, and have found SharePoint to be a wonderful tool for staging malware-based attacks via malicious links and compromised Office 365 accounts to launch attacks on third-party targets. Kalember said a PDF-based phishing campaign associated with Hurricane Michael actually pointed to pages hosted on Azure blob storage.

2. **Subject To Lots Of Identity-Based Attacks** "Microsoft has moved its on-premise identity tools to the cloud, which it pushes heavily to be used around Azure", said Bitglass CTO Anurag Kahol. Organizations typically use active directory from a CASB (cloud access security broker) tool to provide

identity protection around AWS, but in Azure, Kahol said businesses typically end up using Microsoft's identity tools for their entire company. People from different countries attempt to provision attacks against Azure by trying to use an organization's tenant ID and passwords across all sites, Kahol said. As a result, Kahol said he's seen more identity-based attacks against Azure than AWS.

3. **More Frequently Targeted With Malware** Malware has been a big problem for Windows since it's an obvious way to gain control over a machine, which has resulted in Microsoft being a frequent target, according to Aditya Joshi, Threat Stack's EVP of products and technology. Microsoft has an anti-malware offering that integrates with the Azure Security Center, Joshi said, and third-party anti-malware tools can address the issue as well.

2.2 Vulnerabilty Scans and Penetration Testing

In the shared responsibility model of Amazon, their policy permits the customer to test User-Operated Services, i.e. resources created and configured by the user. As an example, AWS EC2 instances can be fully tested, except for attempts to disrupt business continuity, such as trying to launch Denial of Service (DOS) attacks. However, AWS managed systems or their infrastructure is out of the scope of any penetration test performed by customers.

2.2.1 Port Scanning

Port scanning basically continues the information gathering that has started during the reconnaissance phase by identifying open ports and services that are available on the target system. The execution of this step is similar to any penetration test - using cloud services or not, therefore the same tool can be used, that has proved its worth under traditional circumstances. Nmap is a very powerful tool for port scanning in case the suitable flags are applied.

```
-Pn: Treat all hosts as online -- skip host discovery
-p <port ranges>: Only scan specified ports
-sV: Probe open ports to determine service/version info
-v: Increase verbosity level
-A: Enable OS detection, version detection, script scanning, and traceroute
-sS: TCP SYN scan
-T<0-5>: Set timing template (higher is faster)
> nmap -Pn -p 1-65535 -sV -v -A -sS -T4 testforthesis.com
Starting Nmap 7.60 ( https://nmap.org ) at 2018-09-24 11:09 UTC
Nmap scan report for testforthesis.com (99.99.99.91)
Host is up (0.13s latency).
Other addresses for testforthesis.com (not scanned): 88.88.88.81
rDNS record for 99.99.99.91:
ec2-99-99-99-91.eu-east-1.compute.amazonaws.com
Not shown: 65533 filtered ports
PORT STATE SERVICE VERSION
80/tcp closed http
443/tcp open ssl/http nginx
| http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: nginx
Running (JUST GUESSING): Linux 3.X|2.6.X|4.X (90\%), Fortinet FortiOS 5.X
(85\%) OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:2.6
cpe:/o:linux:linux_kernel:4 cpe:/o:fortinet:fortios:5.0.6
Aggressive OS guesses: Linux 3.2 - 3.8 (90\%), Linux 2.6.32 - 3.0 (86\%),
Linux 3.11 - 4.1 (86\%), Fortinet FortiOS 5.0.6 (85\%)
No exact OS matches for host (test conditions non-ideal).
Nmap done: 1 IP address (1 host up) scanned in 949.22 seconds
```

Figure 2.1: Nmap port scanning

As a result, the open and close ports are returned with the recognized services running on them, along with their version. The output also includes supported http-methods and the assumed OS type. Nmap also has recognized the two IP addresses and the EC2-specific host name as well.

2.2.2 Vulnerabilty scan Amazon S3 to find open buckets

Once an attacker has discovered that AWS services are used behind the application, scanning can also be extended to Amazon services. One possible direction is to assume that S3 buckets are also included in the picture. In the following, two tools are presented that can help to find potentially open buckets associated with the target.

1. **Sandcastle bucket enumeration** Sandcastle is a tool for AWS S3 bucket enumeration, written in Python. The script used the name of the target and a wordlist to check whether any buckets can be found associated with the target's name. Based on the status code that is returned when trying to access a bucket, it is clear whether a certain bucket exists and is readable, exists but denies access or does not exist at all.

2. **S3 bucket database** Another helpful tool related to the Amazon S3 service is the online database by Grayhatwarfare [22]. The database currently contains information about 80,000 open buckets and approximately 200 million files. One can search for words of interest and browse the content of the files using the web interface. The purpose of the website is to raise awareness on the open

bucket issue. In case certain files or bucketnames are found that cause any harm, they will be removed after contacting the developers. Compared to the Sandbox bucket enumeration tool, the interface gives more freedom regarding the keywords, thus making the search more customized. However, the database is only updated manually by the maintainers, therefore it might not contain all the current information, whereas the Sandbox tool always returns up-to-date results.

2.2.3 Scanning EC2 instance Metadata using MetaSpoilt

1. **Gather AWS EC2 Instance Metadata** One of Metasploit's post-exploitation module is the Gather AWS EC2 Instance Metadata [23] module which attempts to connect to the AWS EC2 metadata service and crawl and collect all metadata known about the session'd host [24]. The session is required to be a Meterpreter session. Meterpreter is a payload within the framework that provides control over an exploited target system, running as a DLL, loaded inside of any process on the target machine.

   First, the established session has to be put in the background and be upgraded to a Meterpreter session, since this Metasploit module can only work with Meterpreter sessions. This was achieved using the shell to meterpreter post-exploit module. It is possible to check the active sessions within Metasploit, which can help to set the session to the right Id number.The last step is to run the aws ec2 instance metadata module itself. It is only necessary to set

the session to the upgraded Meterpreter session's Id.

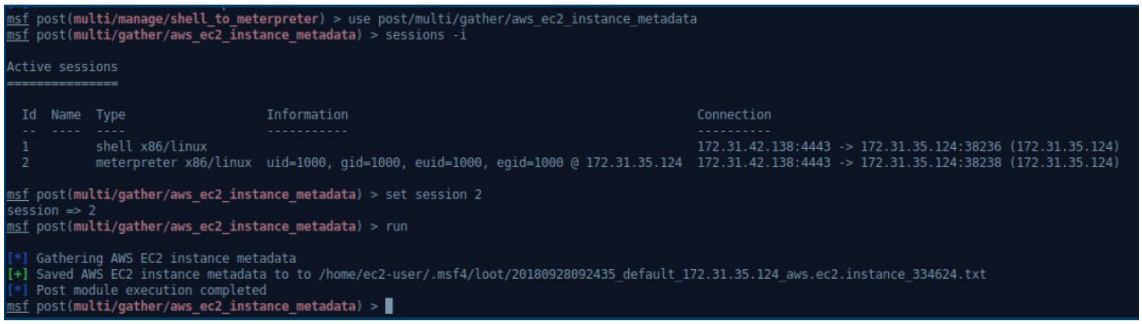

Figure 2.2: Gather Metadata Exploit

As seen in Figure 2.2, the result of the exploit is saved into a text file which contains all the available information on the metadata server. Apart from IAM and security credentials, one can also find the temporary access key, secret key and token. [25]

As seen above, this chapter discussed about the recent/present work performed to cloud services. In the next chapter, I discuss about the my approach to evaluate the in-house cloud security tools in AWS and Azure cloud service providers.

Chapter 3: Contribution

3.1 Overview

It is the customer's responsibility to install the latest security patches for the cloud environment. Vulnerability scanners on traditional networks do not work as efficiently on cloud networks as on local area networks and may not find all crucial vulnerabilities. Administrators should make sure all security patches are up to date and use the tools available to detect potential vulnerabilities. Vulnerability testing requires a strong security background and the highest level of trustworthiness. Even the best automated vulnerability tools produce misinterpreted alarms that are prevented by other actions. An environment can have two or more vulnerabilities that have a lower severity level than one high-level vulnerability, but when combined create a more serious threat to the organization. Some vulnerabilities may remain undetected by the tool, which may present a risk of exploitation. Zhang (2017) [26] reports that Amazon EC2 on Windows and Linux operating systems contains outdated software with critical vulnerabilities. Amazon EC2 cloud has several Common Vulnerabilities and Exposures (CVE) in public images.

In this chapter I created a Virtual Private cloud in us-east region with one public subnet and one private subnet consisting of two ec2 linux instances in each

subnet.

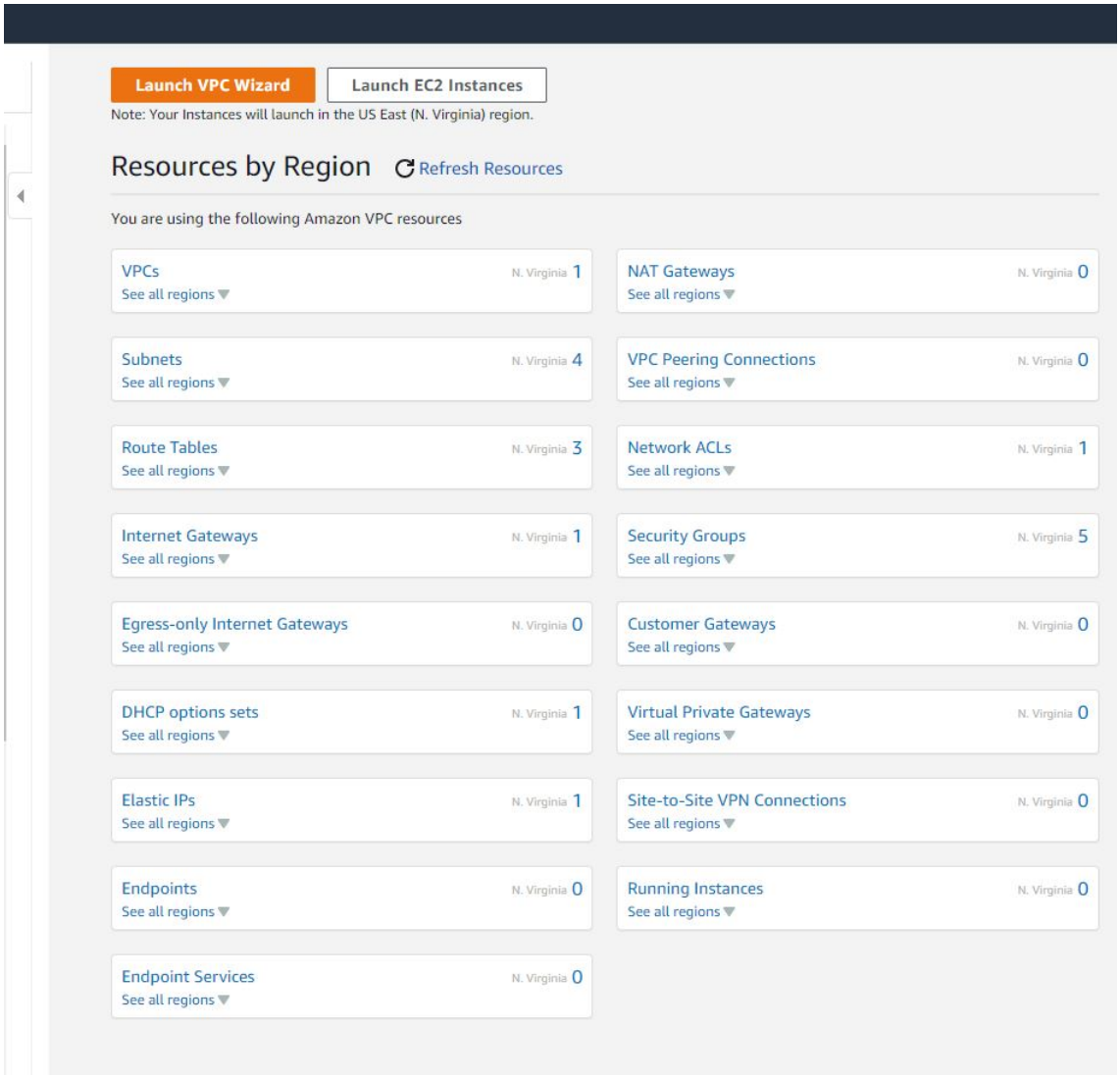

Figure 3.1: VPC Dashboard

Amazon Virtual Private Cloud (Amazon VPC) [27] is a virtual network like the traditional network where you can run your own AWS resources. Its significant advantage is its scalable infrastructure. Traffic between resources and VPC remains internal to the Amazon network and is invisible to outside network.

This chapter also describes the method used to evaluate the suitability of

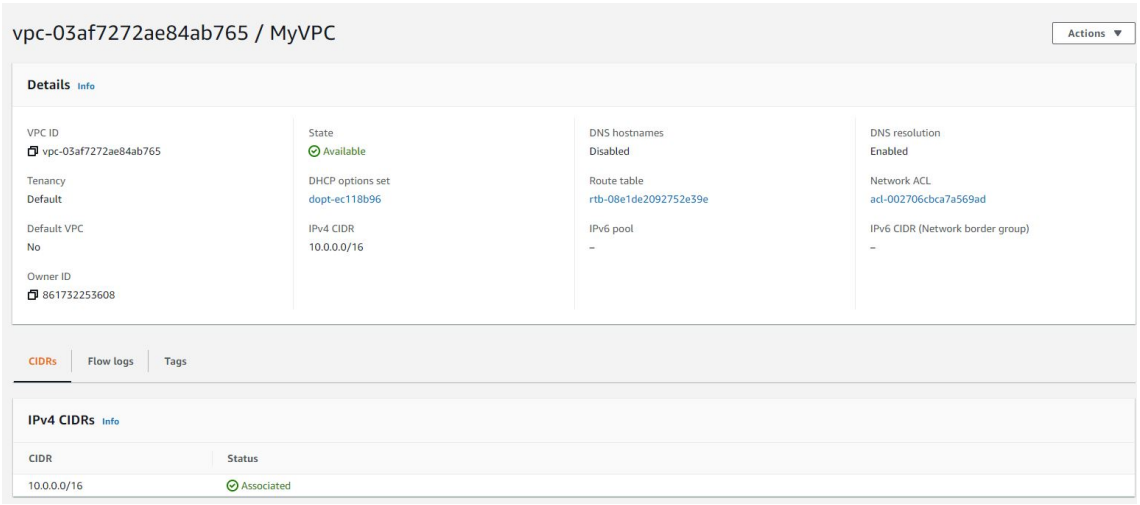

Figure 3.2: Virtual Private Cloud

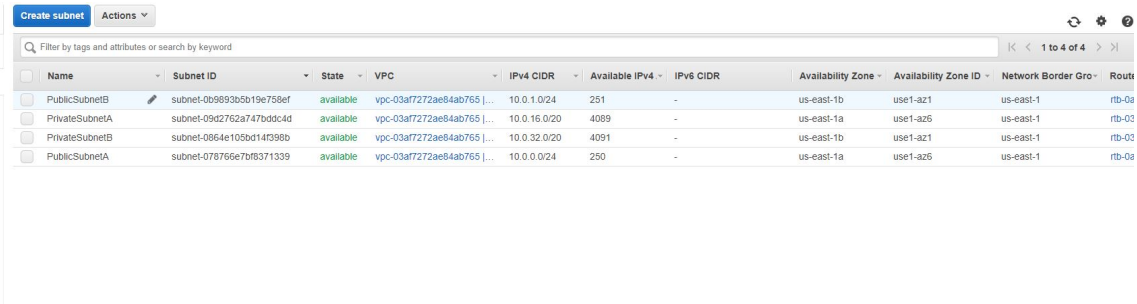

Figure 3.3: Subnets in VPC

Amazon Inspector and Azure Security Center to perform a vulnerability scan on the instances in the VPC's. The detailed results of the reports were excluded from this thesis because the goal was to evaluate the tools to scan the vulnerabilities. The study addresses only significant findings from the reports, sorted by severity level, and some details of the findings.

3.2 Experiments and Data

3.2.1 Scanning tool for AWS

Amazon has detailed instructions for customers to take security tests. For vulnerability scanning, Amazon offers its own service, Amazon Inspector. Amazon maintains and develops this service, which guarantees its continuity and supports its deployment within the organization.

The idea of Amazon Inspector is similar to Nessus [28] vulnerability scanner. In both the agent is installed into target instance where it performs a scan. It produces results according to the selected rules package. The Amazon Inspector security agent is installed on the virtual machines, where it monitors the operation of the instance from inside. The agent uses the CVE database to identify threats and investigates potential system vulnerabilities and security threats. It is not an issue on this case because scanning is first run on pre-production instances, which gives comparable reports to compare with production instances. This requires that necessary vulnerability fixes are conducted in pre-production instances before provisioning to production.

Amazon Inspector Rules Packages Amazon Inspector uses Rules Packages to define performed tests. These Rules Packages are:

- Network Reachability
- Security Best Practices
- CIS Operating System Security Configuration Benchmarks
- Common Vulnerabilities and Exposures

Network Reachability Shows findings about the ports that are reachable from the internet through an internet gateway. This rule can help if ports are misconfigured at the security group level. These scans do not require AWS Inspector Agent, so these can be considered more like an external scan.

Security Best Practices This is a set of certain rules which Inspector will check against and report of them. Some of the best practices include the following

- Disable root login over SSH
- Support SSH version 2 only
- Disable password authentication Over SSH
- Configure password maximum age
- Configure password minimum length
- Configure password complexity

- Enable ASLR
- Enable DEP
- Configure permissions for system directories

CIS Operating System Security Configuration Benchmark This rule checks the operating system against the CIS benchmarks [29] to verify whether the server is following all the best practices mentioned in the CIS Benchmarks. This is a set of certain rules which Inspector will check against and report of them.

Common Vulnerabilities and Exposures Inspector rule CVE basically scans all the packages which are installed in the operating system and it verifies if an associated version has any vulnerabilities. If any vulnerabilities are found, they are listed with details in an Inspector console classified by their severities.

Common Vulnerability Scoring System The Common Vulnerability Scoring System (CVSS) can be used to report the severity of a vulnerability. CVSS consists of three metrics groups: the Base group represents the unchangeable vulnerability properties, the Temporal group the variable properties, and the Environmental group represents the user environment.

Getting Started with Amazon Inspector Following points explain the working and implementation of Amazon Inspector

- Enable Amazon Inspector

The administrator has granted sufficient privileges for the Inspector to use in order to take the necessary action. As the first action, the Amazon Inspector is opened in Amazon AWS Console and enabled in the Amazon AWS account. The three steps shown in Figure 3.4, Install, Run, and Analyse, are

Figure 3.4: Getting started with Amazon Inspector.

not explained here. The start using the Inspector is commenced by clicking the button 'Get started' On the Amazon Inspector home page, it is possible to schedule assessments, run assessments once, and open advanced setup . These are skipped and taken to the Amazon Inspector dashboard page by clicking Cancel.

On the Amazon Inspector home page, it is possible to schedule assessments, run assessments once, and open advanced setup as shown in Figure 3.5. These are skipped and taken to the Amazon Inspector dashboard page by clicking Cancel.

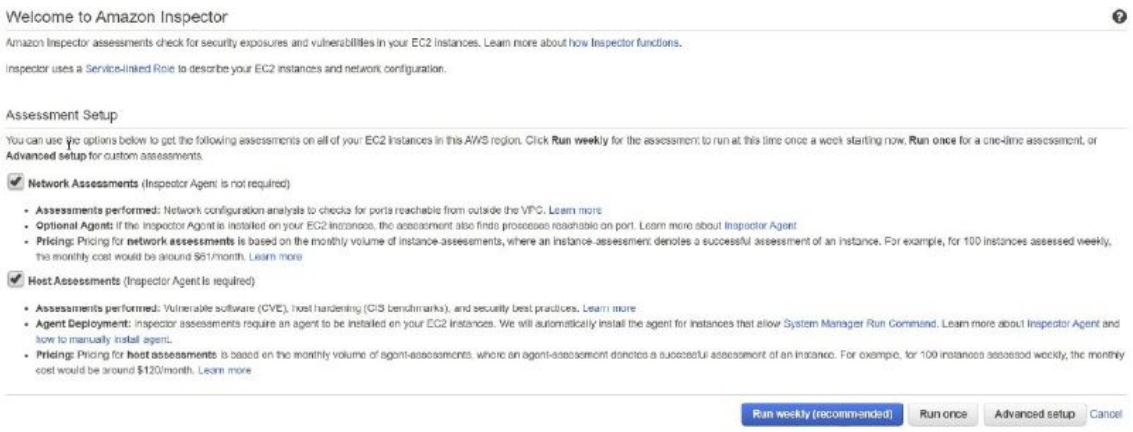

Figure 3.5: Assessment setup options on Inspector.

- Assessment Targets

  Assessment targets is opened by clicking Assessment targets link. By clicking Create button, as shown in Figure 3.6, a new assessment target form is opened. I created assessment targets to define which EC2 instances will be scanned. The option 'All Instances' runs all instances of the AWS account and region. This option is not selected, and the instances are selected manually. The form also has another check box that, when selected, installs the Amazon Inspector Agent on all EC2 instances under this assessment using the Run Command. This option is unchecked, and Inspector Agent is installed manually.

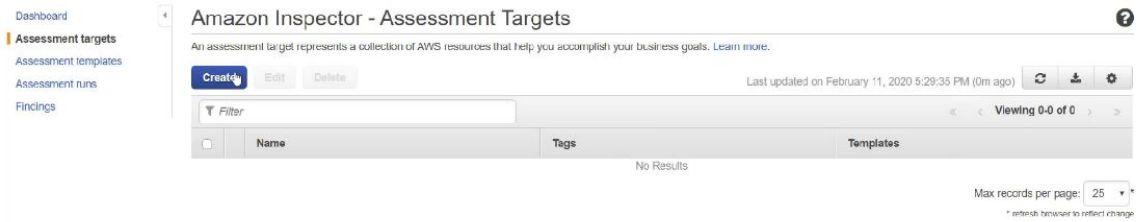

Figure 3.6: Amazon Inspector Target.

- Assessment Template

Assessment templates use an assessment target for tests. Templates define details for tests. The assessment template is associated with an assessment target which is picked on Target name popup field as shown in Figure 3.7 as below

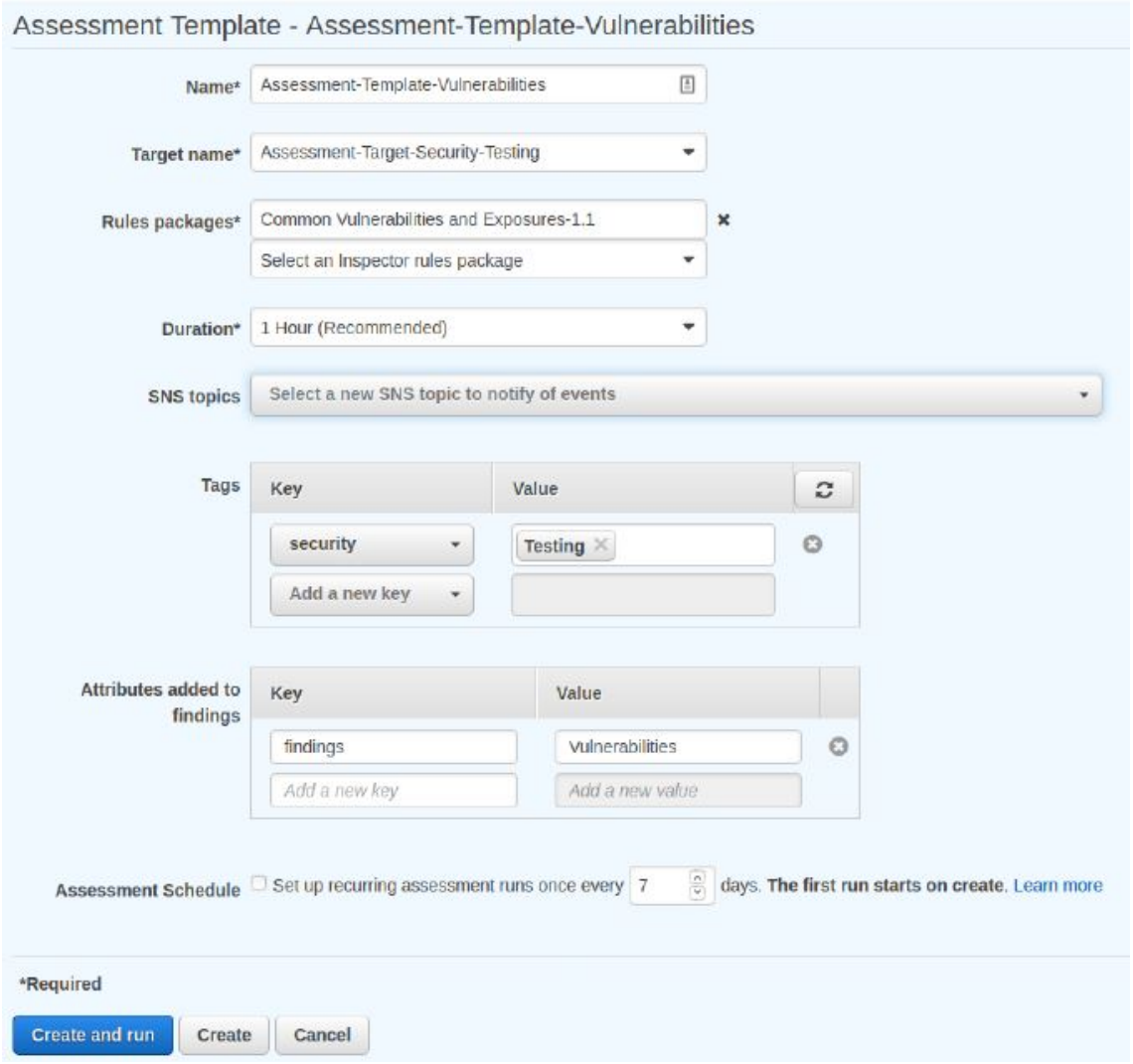

Figure 3.7: Assessment Template

- Findings without Inspector Agent

  One assessment template is created and run for each of the rules packages and

the result is shown in figure “Amazon Inspector – Assessment Templates”. It is remarkable that these runs have performed as they have, and they do not give plenty of results. Only Network Reachability reported few findings, with Low level and 5 Informational level severity as seen in Table 1.

- Install Inspector Agent

  The next step is to install Inspector Agent into EC2 instances. This agent examines EC2 instances from inside and gives a great deal more information about security issues. Amazon Inspector Agent [30] can be installed from inside of EC2 instance. The following is an excerpt of installation process.

  $ curl -O https://inspector-agent.amazonaws.com/linux/latest/install

  Run installation:

  $ sudo bash install

- Findings with Inspector Agent installed

  After agent installation the vulnerability tests were run again, the results were totally different as described in Figure 3.9.

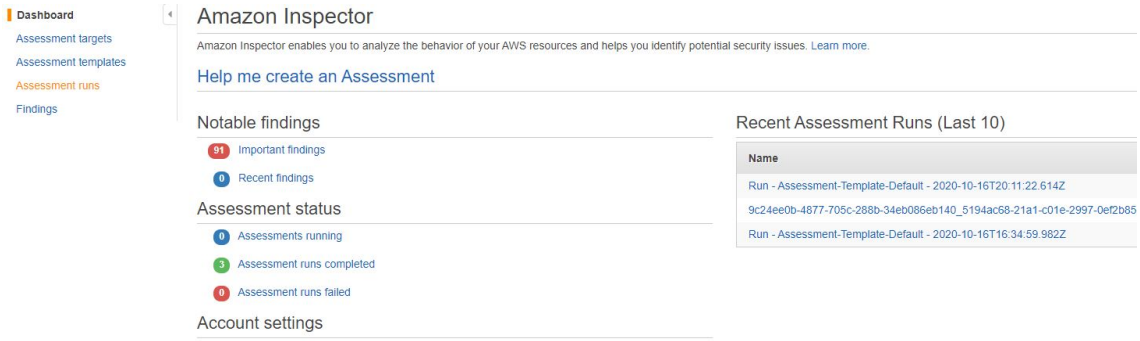

Figure 3.8: Amazon Inspector Findings

About significant findings are in the Vulnerabilities category and in the Benchmarks rules packages category. The total number of significant observations in the Network Reachability and Security-Best-Practices category is very small, only about 4%. This gives a signal to resolve the significance of the observations of the first two categories for their own environment.

The goal was to scan for vulnerabilities from EC2 instances and the network. Amazon Inspector found a lot of significant findings, which was a positive surprise about the tool's operation. On the other hand, it should be noted that if the selected Linux distribution does not support CIS Benchmarks then the result is a failed run, which gives the user the wrong signal.

3.2.2 Excerpt of findings from Amazon Inspector

This section explains about the findings in details. The details of the finding include the following:

- Name of the assessment target that includes the EC2 instance where this finding was registered.
- Name of the assessment template that was used to produce this finding.
- Assessment run start time.
- Assessment run end time.
- Assessment run status.
- Name of the rules package that includes the rule that triggered this finding.

- Name of the finding.
- Severity of the finding.
- Native severity details from the Common Vulnerability Scoring System (CVSS). These include CVSS vector and CVSS score metrics (including CVSS version 2.0 and 3.0) for the findings triggered by the rules in the Common Vulnerabilities and Exposures rules package. For details about the CVSS, see https://www.first.org/cvss/.
- Native severity details from the Center of Internet Security (CIS). These include the CIS weight metric for the findings triggered by the rules in the CIS Benchmarks package. For more information about CIS weight metric, see https://www.cisecurity.org/.
- Description of the finding.
- Recommended steps that you can complete to fix the potential security issue described by the finding.

Figure 3.9 explains the vulnerabilty Daemon finding. It also gives the recommendation to resolve this issue and the rules package category it used i.e "CIS Operating System Security Configuration BenchMarks".

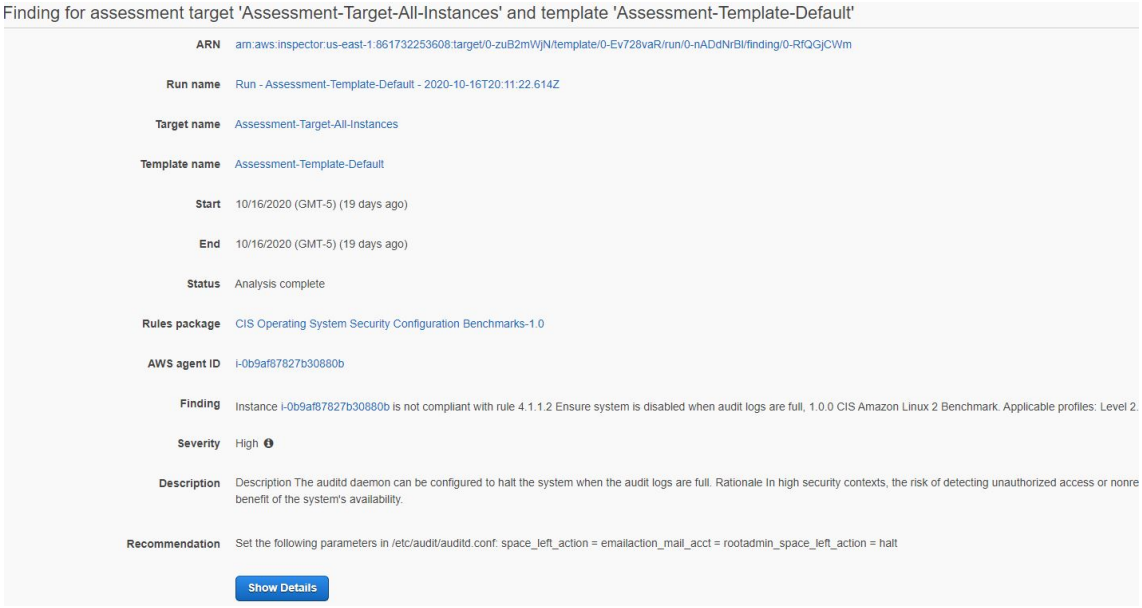

Figure 3.9: Vulnerability Daemon Finding

Figure 3.10 explains the vulnerabilty Port 22 Finding. It also gives the issue that the port is reachable through internet and the recommends to edit the security group. This finding came from "Network Reachablity Rules Package"

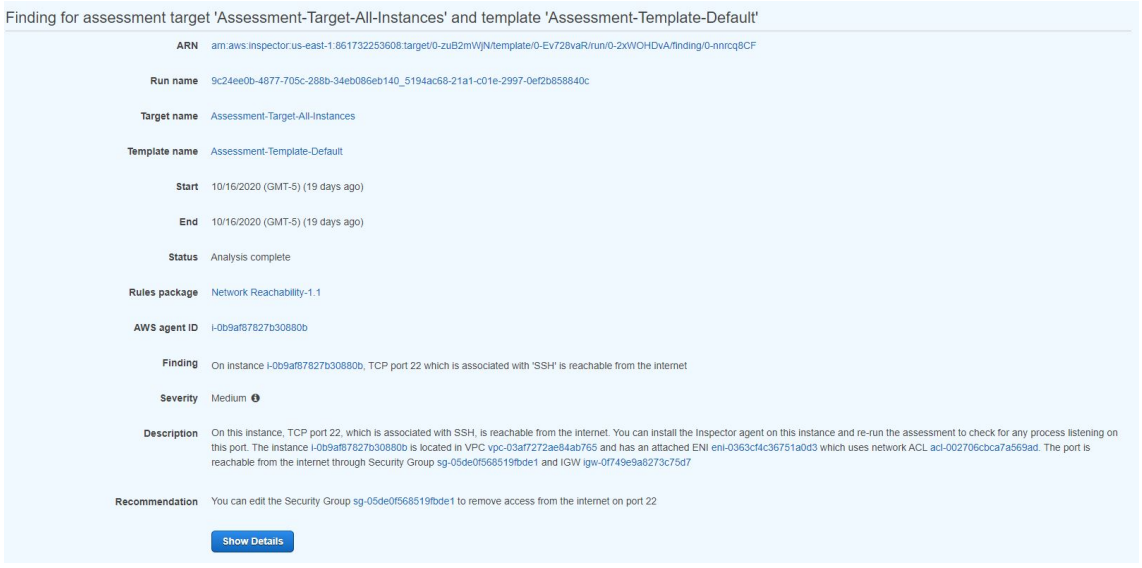

Figure 3.10: Vulnerability Port 22 Finding

Figure 3.11 explains the vulnerabilty Loopback Finding. It also gives the issue that the Loopback traffic is configured and the recommends to implement the loopback rules. This finding came from "CIS rules Package"

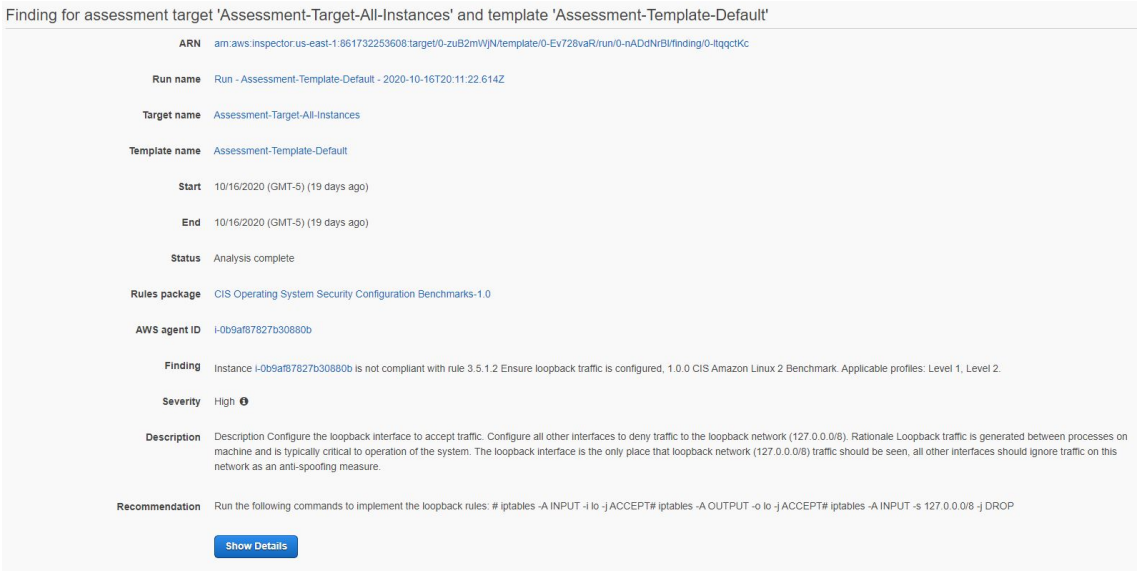

Figure 3.11: Vulnerability LoopBack Finding

Figure 3.12 explains the vulnerabilty SSHGracetime Finding. It also gives the issue that the Gracetime must be set to 60 secs or less. This finding came from "CIS rules Package"

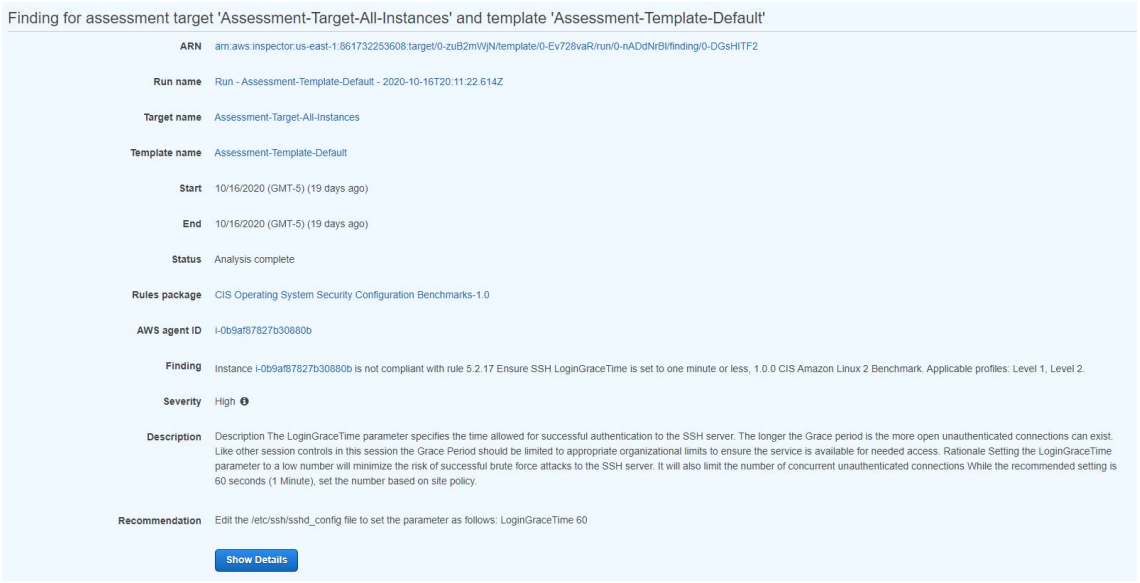

Figure 3.12: Vulnerability SSHGraceTime Finding

Figure 3.13 explains the vulnerabilty SSH Empty Password Finding. It gives the issue that the EmptyPasswordPermit parameter is set to enable and must be disabled. This finding came from "CIS rules Package"

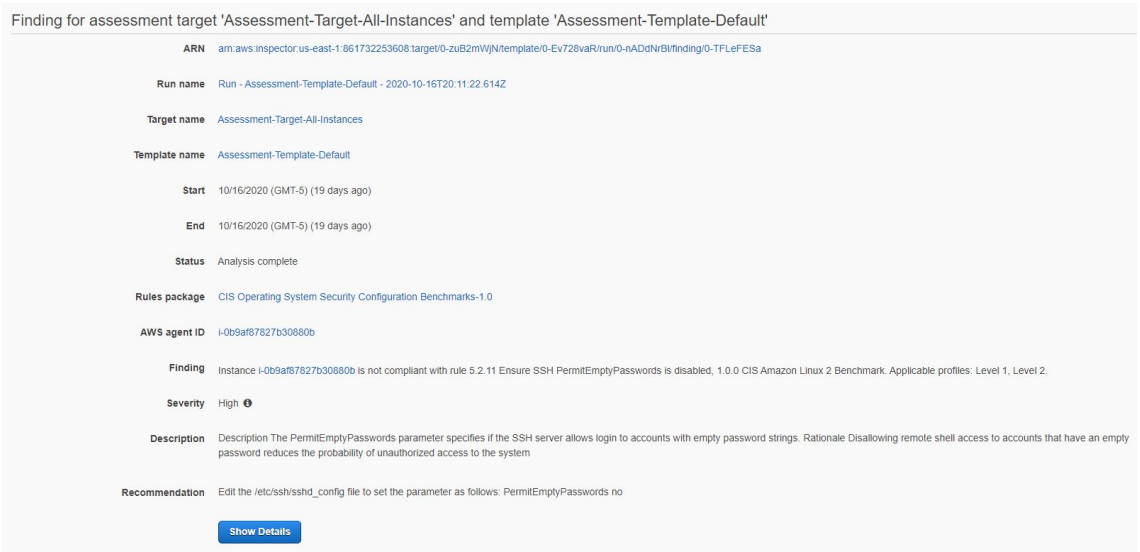

Figure 3.13: Vulnerability SSH Empty Password Finding

3.2.3 Scanning Tool in Microsoft Azure

Azure Security Center's [31] provides built in vulnerability assessment scanner powered by Qualys as part of standard pricing tier. I used Azure Security Center's standard tier for scanning VM's and deployed vulnerability assessment solution powered by Qualys with no additional configuration or extra costs. Qualys's scanner is the leading tool for identifying vulnerabilities. This offering is available to all commercial Azure customers that have enabled Azure Security Center standard pricing tier for VMs.

I followed the following steps below to enable the scanning in Azure

- To make this integration work, a policy named "vulnerability assessment should be enabled on virtual machines" which is part of the "ASC default" initiative was enabled as shown in screenshot below. Upon Azure Policy evaluation, we get the compliance data to identify potential and supported virtual machines which don't have a vulnerability assessment solution deployed.

  "healthy" - VMs that have the extension installed and report data.

  "unhealthy" - VMs which could support the extension, but don't have extension installed

  "not applicable" - Where the OS type/image is not supported

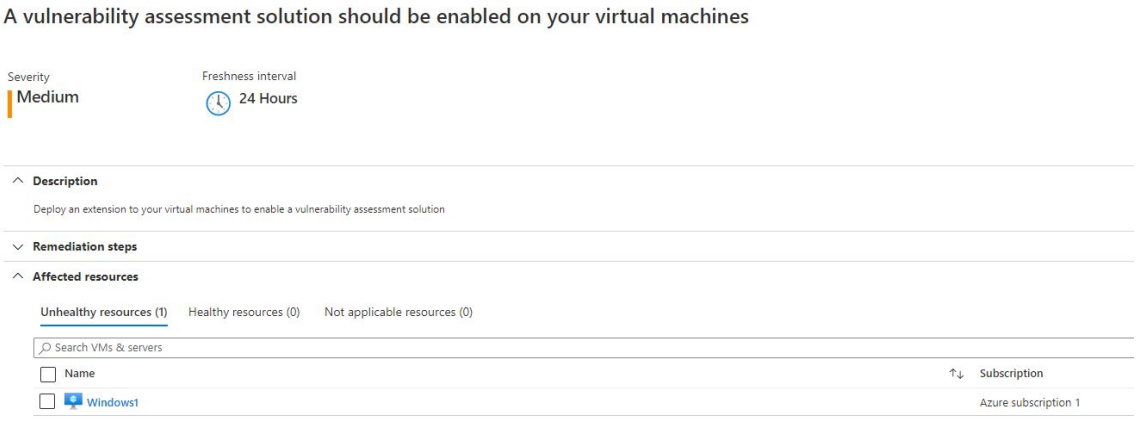

Figure 3.14: Integrate Vulnerability Assessment for your unhealthy VM

- Second, we have to enable the integrated ASC vulnerability scanner by deploying the extension on your selected virtual machine.

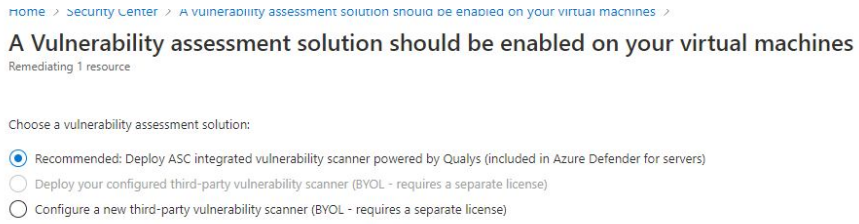

Figure 3.15: ASC enabled

- This agent gathers all the data like network posture, open ports, installed software, registry info, patches installed, environment variables, operating system version, and metadata associated .These scans occur every 4 hours and are performed per VM, where data are collected and sent for analysis to the Qualys Cloud service in the given region. The sent artifacts are considered as metadata and the same as the ones collected by Qualys [32] standalone cloud agent Microsoft doesn't share customer details or any sensitive data with Qualys.

- Qualys analyzes the metadata, registry keys, and other information and builds the findings per VM. Findings are sent to Azure Security Center matching customer's ID and are removed from the Qualys Cloud.

- Findings can be found under "Vulnerabilities in virtual machines should be remediated" recommendations. This recommendation is divided to the affected resources and security checks.

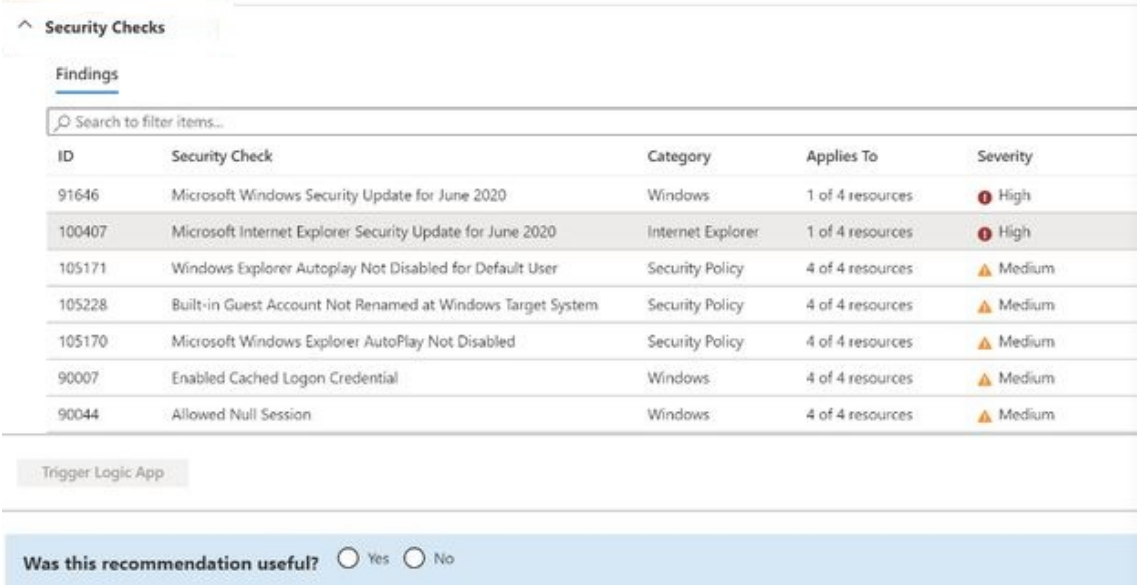

Figure 3.16: Findings showing security checks to be resolved for one of the VM's

Once a security check is selected , window containing the vulnerability name, description, the impact on your resources, severity, if this could be resolved by applying patch, the CVSS base score (when the highest is the most severe one), relevant CVEs is shown. We can also find the threat, remediation steps, and the affected resource. Once the threat is remediated on the affected resource, it will be removed from the recommendation page.

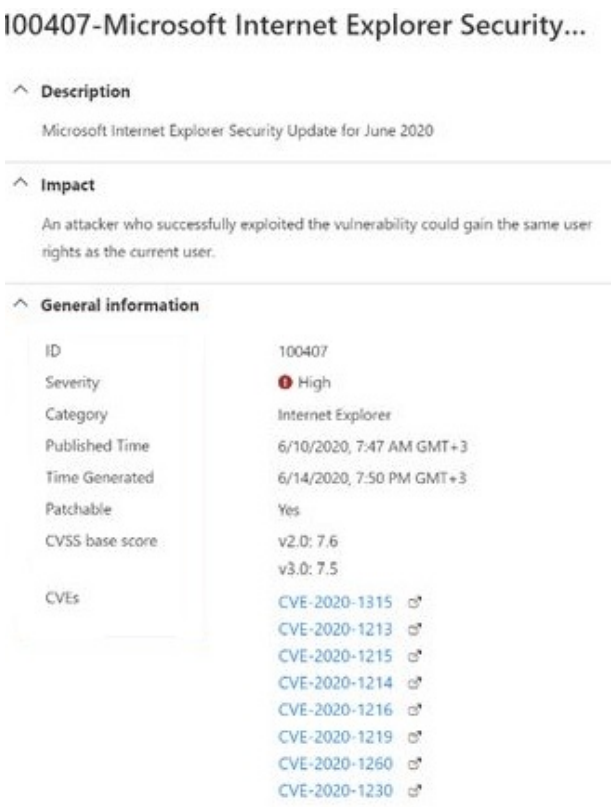

Figure 3.17: Internet Explorer update findings in details

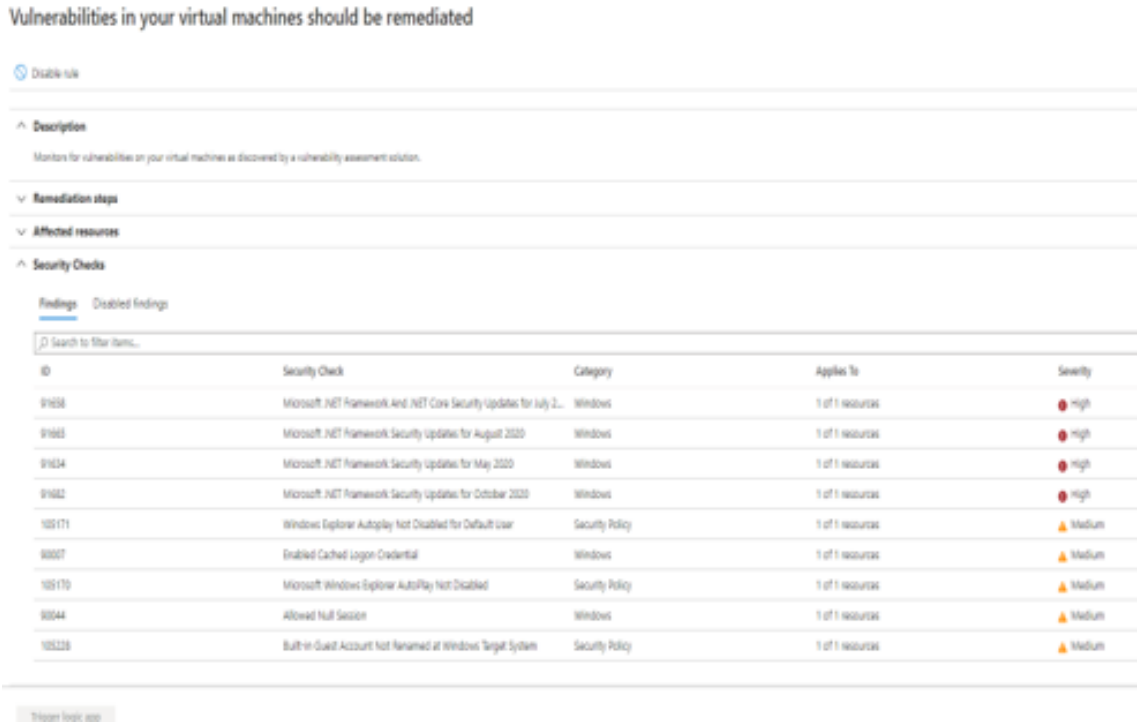

Figure 3.18: Findings showing security checks to be resolved for one of the VM's

Figure 3.19: .net Framework Security Update Missing in details

In this section, we discussed about the tools and the implementation with their results. The next chapter discusses about the interpretation from the findings and evaluate the tools and compare the efficiency of both the tools to scan the vulnerabilities.

Chapter 4: Results and Findings

Vulnerability testing [33] requires a strong security background and the highest level of trustworthiness. Even the best automated vulnerability tools produce misinterpreted alarms that are prevented by other actions. An environment can have two or more vulnerabilities that have a lower severity level than one high-level vulnerability, but when combined create a more serious threat to the organization. Some vulnerabilities may remain undetected by the tool, which may present a risk of exploitation

4.1 Lessons Learned

The number of threats is enormous and should be taken seriously. Vulnerability management is limited in this study.

- As seen from the findings section in above chapters for Amazon Inspector, it gave a large amount of information in reports, detailing of resources and issues found. The report created is rich and comprehensive, it identifies the resource tested and issues it is associated with. The contents expose the threat and also provides guidance on how to fix it. This helps the AWS cloud resources secure and protect from security vulnerabilities. Azure Security Center helps prevent,

detect and respond to threats that can compromise the security of Azure resources. Vulnerability assessments performed by the built-in integration is only available through Azure portal and for Azure apps. It also provided threat report and recommendations to resolve those threats.

- Similarities between AWS and Azure Scanning tools
    - Both provides free-tier usage for 90 days
    - Both Inspector and Azure Security Center are agent based security assessment service
    - Integrated with partner solutions to provide enhanced security assessment and recommendations
    - Security Best practices is a common area of focus though the recommendations vary among them
- Findings from analysing Azure's Security Center Scanning tool
    - The time for scanning cannot be controlled but Azure claims it happens daily
    - Thread remediation is just on click, no need to manually resolve those issues
    - Azure provides an option to integrate with any other partner solution such

- Basic security" is provided for free by default which ensures the must-have security recommendations are addressed for Azure resources without any cost
- Focuses more on Windows OS and SQL DB resources
- Threat prevention can be performed on Firewalls, SQL databases, Storage Disks as well
- Prevention and Threat detection is controlled through policies

- Findings from analysing AWS Inspector Scanning tool
  - The time for scanning can be controlled while configuring AWS Inspector
  - Here, we need to analyze the recommendation and manually resolve the issues
  - No integration with any other partnered solution.
  - Basic security" is provided for free with recommendation addressed with no cost
  - Focuses more on Linux VM's
  - Prevention and Threat detection is controlled through Rules packages as discussed in Chapter 3.

This chapter analyzed both the tools and their findings with similarities and differences which will help to decide which tool one should adopt. The next chapter discusses about the conclusion and future work for this study.

Chapter 5: Conclusions and Future Work

5.1 Conclusions

As seen from the lessons learned in Chapter 4, both AWS Inspector and Azure Security Center have their own strengths and weaknesses. Both of these security assessment services prioritizes providing secure and protected environment to their customers.

Deploying Amazon Inspector as a first step in security best practices is a good starting point for deploying other services because it exposes potential threats and vulnerabilities in instances of the organization. Amazon Inspector is part of the Security Hub, hence, the next best practice step could be to deploy Security Hub and its associated services. Managing reports generated by Inspector is a challenging task and managing vulnerabilities in them is also a great option for the next step in improving security.

The large number of issues identified by Inspector needs to be brought to the attention of project managers, product owners, and others to incorporate security testing into the software development process rather than invest a new dedicated team to achieve the result. However huge delay to support new operating system versions and the overestimation of the severity of findings of the CIS Benchmark

strongly limit its usefulness. But the features available now are sufficient to provide basic security to some resources with very reasonable cost compare to Nessus or Qualys. It is also great for running on machines that are in production as it wont cause extra CPU or Network usage as compare to Nessus or Qualys. As of now, I cannot recommend exclusive use of Inspector as a vulnerability scanning tool, but its worth installing as its cheap and east to use. In future, Amazon will definitely develop more advanced rules and be able deliver good results.

In regards with Microsoft Azure's Security center, to strengthen the security posture of the environment we should definitely consider adopting Azure Security Center in the standard tier, that allows to check in a strict manner all safety criteria and allows to constantly monitor the compliance criteria. The integration in the solution of a vulnerability assessment tool, provided by Qualys, adds further value to the solution, also be able to draw on the knowledge gained by this vendor in the discovery of vulnerabilities. Azure definitely has more features inline with the other vendors at standard pricing rate with an ability to remediate threat by just clicking.

.

5.2 Future Work

The tendency of businesses migrating their services to the cloud is not expected to end in the near future. Amazon is continuously widening the range of their services and offering new opportunities to improve the cloud infrastructure. It also implies emergence of new vulnerabilities, attack platforms and poses additional

security risks.

Vulnerability management provides large amount of data provided by the report. This vulnerability management system could use artificial intelligence to go through and prioritize vulnerabilities. Vulnerabilities that threaten the organization need to be identified to gain an idea of the likelihood of their exploitation and priorities to resolve the issues.

The security logging and monitoring system could collect and categorize the information to be logged into its own categories. This logged information could be tracked through a centralized monitoring system with single dashboard for real-time monitoring of all critical events

Bibliography

[1] Cloud computing timeline. https://pngio.com/images/png-a2072547.html.

[2] Peter Mell and Tim Grance. The NIST definition of cloud computing, 2011.

[3] Amazon AWS well-architected whitepaper. https://d1.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf, 2019.

[4] Global infrastructure. amazon web services, . 16 Apr. 2016.

[5] Amazon web services AWS - cloud computing services, amazon web services, year= 2016.

[6] Comparative study of amazon ec2 and microsoft azure cloud architecture prof vaibhav a gandhi, December, 2013.l.

[7] Azure cloud services by location or region — microsoft azure." azure cloud services by location or region — microsoft azure., 16 Apr. 2016.

[8] Microsoft azure: Cloud computing platform and services., 30 Apr. 2016.

[9] Ronald L. Krutz and Russell Dean Vines. *Cloud Security: A Comprehensive Guide to Secure Cloud Computing*. Wiley Publishing, 2010.

[10] Aws market. https://marketrealist.com/2015/11/can-aws-microsoft-reach-50-market-share-cloud/.

[11] Shared responsibility model. https://aws.amazon.com/compliance/shared-responsibility-model/.

[12] Kpmg: From hype to future: Kpmg's 2010 cloud computing survey. https://cupdf.com/document/cloud-computing-survey-2010-kpmg.html, 2010.

[13] Rosado DG, Gómez R, Mellado D, and Fernández-Medina E. Security analysis in the migration to cloud environments. Future Internet 2012, 4(2):469–487.

[14] Latif S Mather T, Kumaraswamy S. Cloud security and privacy. Sebastopol, CA: O'Reilly Media, Inc.; 2009.

[15] Krishnan. Security and privacy in cloud computing. Master's thesis, Western Michigan University, 2017.

[16] V.Kavitha S. Subashini n. A survey on security issues in service delivery models of cloud computing. Journal of Network and Computer Applications 35(1):1-11, 2011.

[17] Hina Gull Abdullah Alqahtani. Cloud computing and security issues—a review of amazon web services, 2018.

[18] System shock: How a cloud leak exposed accenture's business. https://www.upguard.com/breaches/cloud-leak-accenture.

[19] Upguard. the rnc files: Inside the largest us voter data leak. https://www.upguard.com/breaches/the-rnc-files.

[20] Insurance startup agentrun accidentally leaks customers' personal and health information in cloud configuration error. https://cyware.com/news/insurance-startup-agentrun-accidentally-leaks-customers-personal-and-health-i

[21] Peter benjamin. yas3bl (yet another s3 bucket leak). https://github.com/petermbenjamin/YAS3BL.

[22] How to search for open amazon s3 buckets and their contents. https://buckets.grayhatwarfare.com.

[23] Instance metadata and user data. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-metadata.html.

[24] Rapid7. gather aws ec2 instance metadata. https://www.rapid7.com/db/modules/post/multi/gather/awsec2instancemetadata.

[25] Rizik M. H. Al-Sayyed. An investigation of microsoft azure and amazon web services from users' perspectives, 2019.

[26] T. Zhang. *Detection and Mitigation of Security Threats in Cloud Computing*. PhD thesis, Princeton University, Department of Electrical Engineering, 2017.

[27] What is amazon vpc? https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html.

[28] Nessus. https://resources.infosecinstitute.com/topic/a-brief-introduction-to-the-nessus-vulnerability-scanner/.

[29] Center for internet security. 2019. cis controls v7.1. https://www.cisecurity.org/controls/.

[30] Amazon inspector. https://aws.amazon.com/inspector/.

[31] Securitycenter. https://docs.microsoft.com/en-us/azure/security-center/security-center-introduction.

[32] https://docs.microsoft.com/en-us/azure/security-center/deploy-vulnerability-assessment-vm.

[33] Why your vulnerability management strategy is not working. https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html.